UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/727,744	12/01/2000	Lawrence Richard Fontaine	RA 5312 (1028.1132101)	4178

7590 04/07/2004

Charles A. Johnson
UNISYS Corporation
P O Box 64942
M.S. 4773
St. Paul, MN 55164

EXAMINER	
LI, AIMEE J	

ART UNIT	PAPER NUMBER
2183	7

DATE MAILED: 04/07/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/727,744	FONTAINE ET AL.	
	Examiner	Art Unit	
	Aimee J Li	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 January 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-27 have been considered. Claims 2-27 have been amended as per Applicant's request.

Claim Rejections - 35 USC § 102

2. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

3. Claims 1-18 rejected under 35 U.S.C. 102(b) as being taught by Nakayama et al., U.S. Patent Number 4,788,655 (herein referred to as Nakayama).

4. Referring to claim 1, Nakayama has taught a method for storing a digital value to memory in a pipelined instruction processor, wherein the digital value is read from memory in response to a conditional jump instruction to determine if the condition of the conditional jump instruction is satisfied (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B), the method comprising:

a. Generating at least one status bit based on the digital value to be stored, the at least one status bit indicating if a predetermined condition of a conditional jump instruction is satisfied (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B); and

b. Storing the digital value and the at least one status bit to memory (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

5. Referring to claim 2, Nakayama has taught wherein the conditional jump instruction reads the digital value and the at least one status bit from memory to determine if the condition of the conditional jump instruction is satisfied (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

6. Referring to claim 3, Nakayama has taught wherein the at least one status bit is read from memory at the same time as the digital value (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

7. Referring to claim 4, Nakayama has taught wherein the memory has one or more addressable locations, and the at least one status bit is stored at the same addressable location as the corresponding digital value (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, the sign bit is held at the same address location as the value.

8. Referring to claim 5, Nakayama has taught wherein the at least one status bit is set high if the digital value is zero (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

9. Referring to claim 6, Nakayama has taught wherein the at least one status bit is set high if the digital value is a positive value (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

10. Referring to claim 7, Nakayama has taught wherein the at least one status bit is set high if the digital value is negative (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2,

lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

11. Referring to claim 8, Nakayama has taught wherein the at least one status bit is set high if the digital value is a non zero value (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

12. Referring to claim 9, Nakayama has taught wherein the at least one status bit is set high based on the value of the least significant bit of the digital value (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

13. Referring to claim 10, Nakayama has taught in a pipelined instruction processor that executes instructions including conditional jump instructions, one or more of the conditional jump instructions reading a digital value from memory to determine if the condition of the conditional jump instruction is satisfied (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B), the improvement comprising:

a. Status bit generator for generating at least one status bit based on a digital value, the at least one status bit indicating if a predetermined condition of a conditional jump instruction is satisfied (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B); and

b. Storing means for storing the digital value and the at least one status bit to the memory (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

14. Referring to claim 11, Nakayama has taught wherein a selected conditional jump instruction reads the digital value and the at least one status bit from memory to determine if the condition of the conditional jump instruction is satisfied (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

15. Referring to claim 12, Nakayama has taught wherein the at least one status bit is read from the memory at the same time as the digital value is read (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

16. Referring to claim 13, Nakayama has taught wherein the memory has one or more addressable locations, and the at least one status bit is stored at the same addressable location as the corresponding digital value (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, the sign bit is held at the same address location as the value.

17. Referring to claim 14, Nakayama has taught wherein the at least one status bit is set high if the digital value is zero (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

18. Referring to claim 15, Nakayama has taught wherein the at least one status bit is set high if the digital value is a positive value (Nakayama Abstract; column 1, lines 9-13 and 49-68;

column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

19. Referring to claim 16, Nakayama has taught wherein the at least one status bit is set high if the digital value is negative (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

20. Referring to claim 17, Nakayama has taught wherein the at least one status bit is set high if the digital value is anon zero value (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

21. Referring to claim 18, Nakayama has taught wherein the at least one status bit is set high based on the value of the least significant bit of the digital value (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B). In regards to Nakayama, it does not matter whether the status bit is set high or low.

Claim Rejections - 35 USC § 103

22. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

23. Claims 19-22 and 24-27 rejected under 35 U.S.C. 103(a) as being unpatentable over Watson et al., U.S. Patent Number 3,573,854 (herein referred to as Watson) in view of Nakayama et al., U.S. Patent Number 4,788,655 (herein referred to as Nakayama).

24. Referring to claim 19, Watson has taught in a pipelined instruction processor that executes instructions including conditional jump instructions, the improvement comprising:

a. A plurality of addressable registers, each of the addressable registers (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5);

b. Logic to access a current instruction (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5);

c. A jump look-ahead controller for generating a jump look-ahead signal using the address that identifies one of the addressable registers and the jump field that identifies a jump status bit within the identified addressable register, the jump look-ahead signal is a function of the identified jump status bit (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5);

d. Tracking logic for tracking the addresses of a predetermined number of previous instructions in the pipelined instruction processor and comparing the addresses of each previous instruction to the address of the current instruction to generate a series of jump disable signals (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line

74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5); and

e. Conflict detection logic for generating a jump early signal using the jump look-ahead signal and the series of jump disable signals, the jump early signal initiates the conditional jump depending on the values of the jump disable signals (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5).

25. Watson has not taught:

a. One or more of the conditional jump instructions reading a digital value from memory to determine if the condition of the conditional jump instruction is satisfied;

b. Storing a value that includes a digital value and at least one jump status bit; and

c. Wherein the current instruction includes an address and a corresponding jump field, the address identifies one of the addressable registers and the corresponding jump field identifies a jump status bit of the at least one jump status bits within the identified addressable register.

26. Nakayama has taught:

a. One or more of the conditional jump instructions reading a digital value from memory to determine if the condition of the conditional jump instruction is satisfied (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B);

b. Storing a value that includes a digital value and at least one jump status bit (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B); and

c. Wherein the current instruction includes an address and a corresponding jump field, the address identifies one of the addressable registers and the corresponding jump field identifies a jump status bit of the at least one jump status bits within the identified addressable register (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

27. A person of ordinary skill in the art at the time the invention was made would have recognized that the condition code generating system of Watson requires less time to process and complete, thereby increasing the speed of the processor. Therefore, it would have been obvious to a person of ordinary skill in the art at the time this invention was made to incorporate the condition code generating system to increase the speed of the processor.

28. Referring to claims 20 and 21, Watson has not taught

a. Wherein each jump status bit is dependent on the digital value stored in the corresponding addressable register; and

b. A bit status generator for generating the corresponding jump status bits.

29. Nakayama has taught:

a. Wherein each jump status bit is dependent on the digital value stored in the corresponding addressable register (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B); and

b. A bit status generator for generating the corresponding jump status bits (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

30. A person of ordinary skill in the art at the time the invention was made would have recognized that the condition code generating system of Watson requires less time to process and complete, thereby increasing the speed of the processor. Therefore, it would have been obvious to a person of ordinary skill in the art at the time this invention was made to incorporate the condition code generating system to increase the speed of the processor.

31. Referring to claim 22, Watson has taught a prediction logic block responsive to the jump early signal for implementing a prediction algorithm to predict the conditional jump depending on the values of the jump disable signals (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5).

32. Referring to claim 24, Watson has taught wherein the predetermined number of instructions are sequentially piped through an execution pipeline after being piped through a pre-fetch pipeline, the execution pipeline includes a write-back stage (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5).

33. Referring to claim 25, Watson has taught wherein the addressable register is written during the write-back stage (Watson column 1, lines 40-63; column 1, lines 56-75; column 4,

lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5).

34. Referring to claim 26, Watson has taught wherein the execution pipeline further includes an address generation stage, a present address stage, an output operand stage, a capture data stage, and an arithmetic operation stage, all before the write-back stage (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5).

35. Referring to claim 27, Watson has taught a method for determine if a condition of a conditional jump instruction is satisfied in a pipelined instruction processor, the method comprising:

a. Generating a jump look-ahead signal that is a function of the selected jump status bit read from the selected address location of the addressable memory, the identified jump status bit is accessed using the address and the jump field of the current instruction (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5);

b. Tracking the addresses of a predetermined number of previous instructions in the pipelined instruction processor and comparing the addresses to the address of the current instruction to generate a series of jump disable signals (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to

column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5); and

c. Generating a jump early signal using the jump look-ahead signal and the series jump disable signals, the jump early signal initiates a conditional jump depending on the value of the jump disable signals (Watson column 1, lines 40-63; column 1, lines 56-75; column 4, lines 37-47; column 4, line 68 to column 5, line 61; column 5, line 74 to column 6, line 37; column 6, lines 69-74; column 7, line 7 to column 8, line 13; Figure 4; and Figure 5).

36. Watson has not taught:

a. Storing a digital value and one or more jump status bits that are based on the digital value in each of a plurality of address locations in an addressable memory; and

b. Accessing a current instruction, the current instruction having an address and a jump field, the address identifies a selected address location of the addressable memory, and the jump field identifies a selected jump status bit of the selected address location.

37. Nakayama has taught:

a. Storing a digital value and one or more jump status bits that are based on the digital value in each of a plurality of address locations in an addressable memory (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B); and

b. Accessing a current instruction, the current instruction having an address and a jump field, the address identifies a selected address location of the addressable memory, and the jump field identifies a selected jump status bit of the selected address location (Nakayama Abstract; column 1, lines 9-13 and 49-68; column 2, lines 20-50; Figure 1A; Figure 1B; Figure 5A; and Figure 5B).

38. A person of ordinary skill in the art at the time the invention was made would have recognized that the condition code generating system of Watson requires less time to process and complete, thereby increasing the speed of the processor. Therefore, it would have been obvious to a person of ordinary skill in the art at the time this invention was made to incorporate the condition code generating system to increase the speed of the processor.

39. Claim 23 has been rejected under 35 U.S.C. 103(a) as being unpatentable over Watson in view of Nakayama as applied to claim 19 above, and further in view of Heuring and Jordan's Computer Systems Design and Architecture ©1997 (herein referred to as Heuring).

40. Watson in view of Nakayama has not taught wherein the tracking logic includes a queue for sequentially storing a pre-determined number of instructions prior to sequentially piping the pre-determined number of instructions through a read stage and decode stage in a pre-fetch pipeline. Heuring has taught wherein the tracking logic includes a queue for sequentially storing a pre-determined number of instructions prior to sequentially piping the pre-determined number of instructions through a read stage and decode stage in a pre-fetch pipeline (Heuring Pages 92-95). A person of ordinary skill in the art at the time the invention was made would have recognized that pre-fetching increases the speed and efficiency of the processor. Therefore, it

would have been obvious to a person of ordinary skill in the art at the time this invention was made to incorporate the pre-fetching of Heuring to increase processor speed and efficiency.

Response to Arguments

41. Examiner withdraws objections to the drawings in favor of the amended drawings.

42. Examiner withdraws objections to the title in favor of the new title.

43. Examiner withdraws objections to the claims in favor of the amended claims

44. Applicant's arguments filed 23 January 2004 have been fully considered but they are not persuasive.

45. Applicant argues in essence on pages 15-18 and 21

> "Nakayama et al. do not appear to relate in anyway to a method for storing a digital value to memory in a pipelined instruction processor, wherein the digital value is read from memory in response to a conditional jump instructions to determine of the condition of the conditional jump instruction is satisfied, as suggested by the Examiner (Emphasis Added). Furthermore, Nakayama et al. do not appear to generating at least one status bit based on the digital value to be stored, the at least one status bit indicating if a predetermined condition of a conditional jump instruction is satisfied, or storing the digital value and the at least one status bit to memory..."

46. This has not been found persuasive. Nakayama has taught "...it is necessary to set the external condition of the binary floating point data, which is obtained as a result of the arithmetic operation, in a status register which can be checked by software (Nakayama column 1, lines 9-13)." A conditional branch, also known as a conditional jump in the art, is a software instruction

which checks the status register before transferring program control to another instruction. The condition field set in Nakayama, which indicates whether a certain condition is satisfied or not, is used by software, specifically a software conditional branch instruction. Please see the attached references for further definitions of "conditional branch", "branch", and "conditional jump".

47. Applicant argues in essence on pages 18-20, "Nowhere do Watson et al. disclose or suggest a jump status bit, and in particular, a jump status bit that is accessed using the address and the jump field of the current instruction." This has not been found persuasive. Watson has taught in column 7, line 7 to column 8, line 13 how conditional branches are handled, including accessing a jump status bit, also known as the condition. The cited lines allude to accessing the condition via an address and the jump field. It is inherent, as well, that the condition would be accessed using an address and the jump field, since that is the only way the system would know where to look in memory and what condition to specifically check for.

48. Applicant argues in essence on page 20, "...Applicants do no see where in Watson et al. 'a series of jump disable signals' are suggested. Also, Applicants do not believe Watson et al. disclose or suggest generating a jump early signal using a jump look-ahead signal and a series of jump disable signals." This has not been found persuasive. Watson has described in column 7, line 7 to column 8, line 13 how jump signals are disabled when the condition is not found to be true. Specifically, Watson has taught "...When the line 234e is in the zero state the condition is not satisfied and the program loop will be followed. However, when the output of the flip-flop 234d causes line 234e to be in one state, the decode branch unit 234 is inhibited so that there will be no signal on line 234a." This means that, when a jump condition is not satisfied, the jump signals are disabled, so the jump will not be taken. Watson has also taught a jump look-ahead

mechanism in column 1, lines 45-54 and column 6, lines 4-37. The jump look-ahead device responds to a look-ahead instruction, which functions as a signal, to enable or disable the jump signals.

Conclusion

49. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

50. A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

51. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Aimee J Li whose telephone number is (703) 305-7596. The examiner can normally be reached on M-T 7:30am-5:00pm.

52. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

53. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

AJL
Aimee J. Li
April 5, 2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100